

**Boston University**

**Electrical & Computer Engineering**

EC464 Senior Design Project

**Second Prototype Test Report**

*Hybrid Battery Pack for Self-Launching eGliders*



Team #23

Team Members

Spencer P. Piligian, [spiligia@bu.edu](mailto:spiligia@bu.edu)

Daniel Kao, [dkao1234@bu.edu](mailto:dkao1234@bu.edu)

Yusheng Chen, [chenyush@bu.edu](mailto:chenyush@bu.edu)

Stephen Wang, [swang3@bu.edu](mailto:swang3@bu.edu)

Ismaeel AlAlawi, [ialalawi@bu.edu](mailto:ialalawi@bu.edu)

**I. Required Materials**

*Hardware:*

* 3 x LithiumWerks A123 P6 (3.3V, 2500 mAh) lithium iron phosphate (LiFePO4) rechargeable battery cells
* 2 x Molicel P42A (3.6 V, 4200 mAh) lithium-ion (Li-ion) rechargeable battery cells
* 3 x BeilaMoo 26650 3.7 V single slot battery holders
* Surface mount technology (SMT) KeyStone Electronics 21700/20700 dual battery holder w/ coil springs
* IXYS VUO 190-08 NO7 three-phase diode bridge rectifier
* Littelfuse FL-1 PBT-GF15 7.5A fuse
* TE Connectivity Potter & Brumfield T9GS1L14-12 30A relay
* TeamTriforceUK A50S V2.1 (50A burst) VESC-supported motor controller
* XT30 connector (battery input)
* MR30 connector and cable cover (motor output)
* 2 x pico-clasp connector shells
* T-Motor U-Power U12 KV90 brushless motor
* Adafruit 574 10A red LED panel mount current sensor (ammeter)
* Arduino Nano 33 BLE nRF52840 microcontroller
* 2 x microUSB to USB cables
* Male-to-male jumper wires
* 1 x uxcell negative temperature coefficient (NTC) 10K Ohm thermistor
* Test Products Intl. (TPI) dual temperature thermocouple digital thermometer 343
* 1 x 10K Ohm resistors
* Alligator clips
* Crocodile clips
* Hook clips
* Banana plugs
* Electrical ring tongue terminals AWG 18-22
* 830 point solderless breadboard
* TE Connectivity TE2500B1R0J 50 W - 2500 W chassis-mount power resistor
* AWG 8, 12 & 18 wires
* ACDelco SuperALKALINE 6LR61 9V battery
* PKCELL Extra heavy duty 6F22 9V battery
* 2 x battery snap connectors (9V)
* 12V lab power supply (e.g. Keysight E3631A triple output DC power supply or Tekpower TP3016M portable handheld variable DC power supply)
* Electrical insulation tape and masking tape
* 5 x MCR Safety safety glasses
* Multimeter

*Software:*

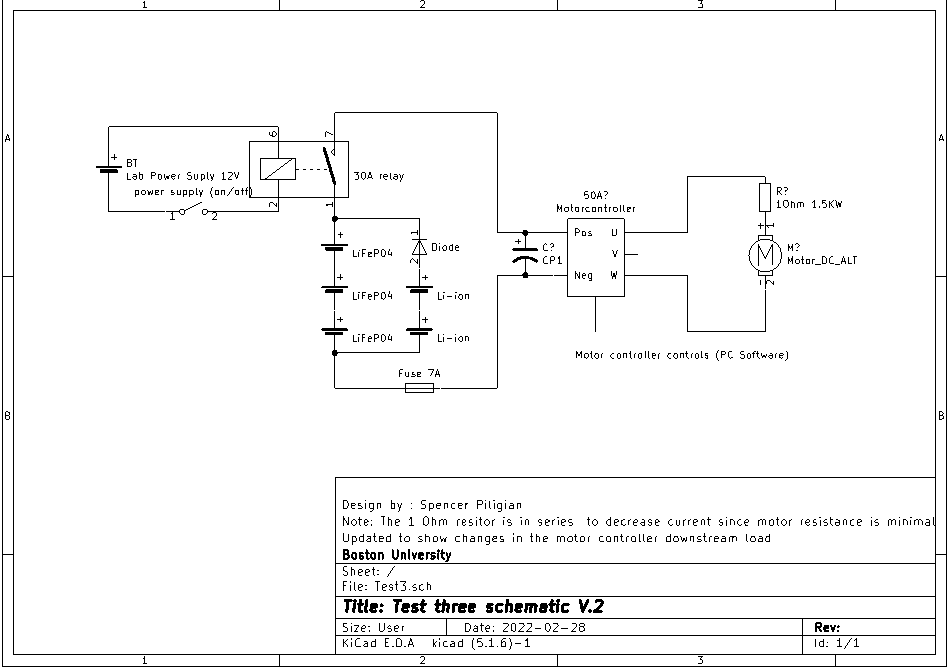
* VESC Tool (free version) GUI developed by Benjamin Vedder
  + Qt Toolkit
  + .zip file extractor (Windows, Linux, Mac OS X, and Android supported)
  + Source code: <https://github.com/vedderb/vesc_tool>
* VESC-Lisp, Common Lisp, or LispBM Scripts
  + ANSI Common Lisp GNU v2.49 (2010-07-07 build) compiler (REPL)
  + For Mac OS X, install via MacPorts
  + For Windows and Linux (Ubuntu, Debian, Sun Solaris, etc.), install directly from <https://clisp.sourceforge.io/>
* Arduino IDE v1.8.19
  + Supported for Windows 7+, Linux (32-bit & 64-bit), Mac OS X (10.10 or newer), ChromeOS
  + Download directly from <https://www.arduino.cc/en/software>
* Arduino C/C++ Scripts (sketches)
  + Firmata library for all boards
  + *LiquidCrystal.h* library – optional for LCD display of temperature logs

**II. Setup**

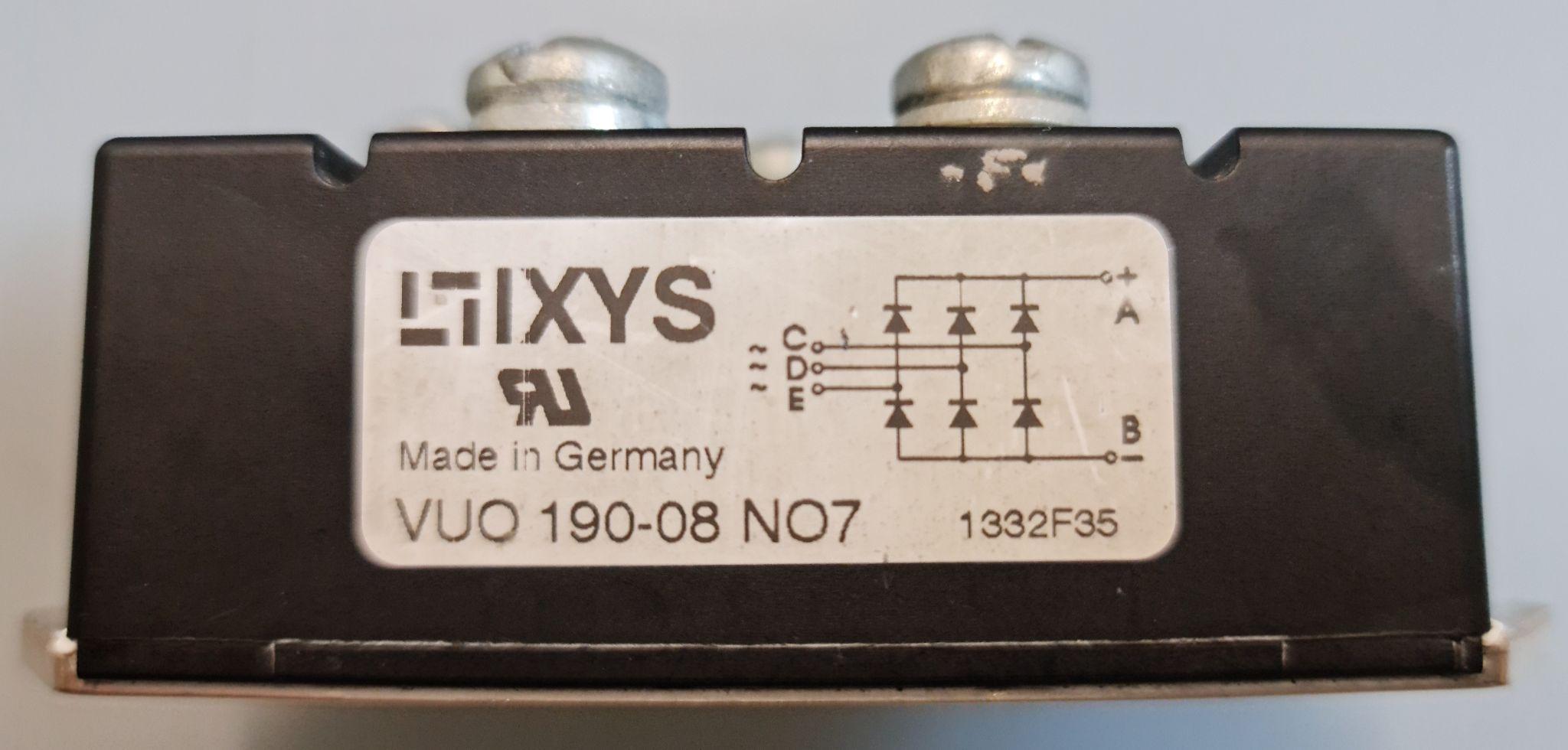
In order to properly conduct this second in-lab prototype test, there are several hardware and software requirements that need to be met. On the hardware side, there are mainly two parts: the battery testing circuitry and the arduino microcontroller-breadboard configuration. In particular, the battery circuit includes a 12 V DC power supply that is connected to a 30 A relay and that relay is connected to a a parallel connection of LiFePO4 and Li-ion batteries in a 3:2 ratio respectively, in which a Schottky barrier diode is placed in series with the Li-ion batteries as it helps prevent reverse leakage current from one battery pack to another. It is important to note that this Schottky barrier is part of a diode bridge rectifier (i.e. terminals are adjusted accordingly in order to retrieve a single Schottky barrier diode from the rectifier circuit). There exists a 7 A fuse that is connected in series with the motor controller and the current sensor for safety standard purposes. The motor controller is properly connected to a motor using connection shells and the motor is then coupled with a power resistor in order to dump energy into it – by connecting the motor in series with the power resistor, the inductance can be tuned correctly very easily. For safety reasons, the arduino microcontroller circuit contains a thermistor that would log the temperature of the batteries; the digital thermocouple thermometer would log the temperature of the MOSFETs in the motor controller. The battery and motor controller voltages will also be recorded appropriately using a multimeter. On the software side, using VESC Tool software, the motor and battery current readings are plotted over time graphically in order to determine the amount of current flowing through the motor that is required for the battery to reach a discharge current of 5A. For the arduino software, the temperatures are logged according to a fixed sampling rate (1 Hz typically) and displayed on the serial monitor.

**III. Pre-testing Setup Procedures**

* *Battery testing circuit hardware side (see* ***Figure 1*** *for visual reference)*
  + Connect 12 V lab DC power supply to the bottom left terminal of the 30 A relay using alligator or crocodile clips
  + Connect 9 V battery to the relay (top left and right connections) by soldering battery snap connectors to power it up and connect other 9 V battery to the current sensor using battery snap connectors to also power it up
  + Place each of the three LiFePO4 batteries in their respective single slot battery holders
  + Place both Li-ion batteries in the dual slot battery holder
  + Solder the wires connecting the LiFePO4 battery holder contacts in series and solder the first ring terminal wire to one of the LiFePO4 battery holders accordingly and the other to the Li-ion dual battery holder to create the parallel connection
  + Connect the ring terminal connected to the LiFePO4 battery holders into terminal ~C (phase-C) of the rectifier and the other terminal (which is connected to Li-ion batteries) to terminal B- (see **Figure 2** for rectifier reference schematic)
  + Solder the LiFePO4 battery holder and rectifier junction into the 30 A relay’s bottom right terminal
  + Solder the 7 A fuse into the bottom-right terminal of the Li-ion battery holder and connect its other end to the negative top terminal of the current sensor
  + Connect the positive top terminal of the current sensor to the motor controller positive XT30 connection; the negative terminal of this connection should be connected to the multimeter accordingly for voltage readings.
  + Connect the positive terminal of the motor controller MR30 connector into the positive terminal of the multimeter to complete the connection for motor controller voltage readings
  + Connect the A50S motor controller to the laptop using micoUSB-USB cable
  + Connect the pico-clasp connector shell to MR30 connector, and connect that to the brushless motor’s connector
  + Connect the positive and negative terminals of the motor controller-motor junction to the positive and negative terminals of the power resistor respectively
* *Arduino Nano microcontroller hardware side*
  + Connect Arduino Nano to computer with microUSB-USB cable (if using a laptop with only USB-C ports, then a USB to USB-C hyperdrive is needed)
  + Since resistors are blind to polarity, the thermistor and resistor ends are arbitrary, so for clarification, let one end be (a) and the other be (b) for both respectively
  + Connect pin #27 (5V DC power) to end (a) of 10K resistor, and connect end (b) of the resistor to end (a) of thermistor using male-to-male jumper wires
  + Connect pin #19 (A0/D14) to end (a) of thermistor using male-to-male jumper wires
  + Connect pin #29 (GND) to end (b) of thermistor using male-to-male jumper wires
* *VESC Tool setup software side*
  + Create an account on <https://vesc-project.com/> > click on “VESC tool’ tab > purchase VESC Tool Free for €0.00 (no billing information required) > click on “Purchased Files” link (top right corner) > download and extract “vesc\_tool\_BETA\_ALL.zip” (or for Windows users, download “vesc\_tool\_free\_windows.zip”)
  + Alternatively install directly via source code: <https://github.com/vedderb/vesc_tool>
  + For Windows and Linux users, Install REPL Lisp compiler via SourceForge.
  + For Mac OS X users install MacPorts v2.7.1
    - Install Xcode and Xcode Command Line Tools
    - Agree to Xcode license in Terminal: sudo xcodebuild -license
    - Install MacPorts according to version of OS X (convenient installation through .pkg installer for version 10.15+) on <https://www.macports.org/install.php>
    - After MacPorts installation, type sudo port install clisp to install common lisp implementation (interpreter, compiler, debugger, etc.).
* *Arduino IDE software side*
  + After installing Arduino IDE as mentioned in the required materials section, go to Tools > Manage Libraries > Drop-down “Type” > Installed. Double-check that Firmata built-in library is uncorrupted and installed properly. For LCD temperature display option, if not already installed, search for LiquidCrystal and install version 1.0.6
  + For microcontroller testing, go to Tools > select Boards Manager > search for Arduino AVR Boards > ensure most up-to-date version installed (v1.8.5 as of in-lab testing) > select Board: Arduino Nano > under same tab, select appropriate serial port when board is connected.
  + Once Arduino board properly connected with aforementioned hardware configuration on breadboard, run the temperature logging Arduino code and check the serial monitor for logged temperature readings – modify sampling rate as appropriate. As a quick test, check that room temperature is properly recorded
* *Diode, 9V batteries, and properly soldered electrical connection testing (if necessary)*
  + Using a multimeter, test the voltages/currents across the terminals of the Schottky junction diode and the 9V batteries, as well as other electrical connections in the circuit to ensure that the connections aren’t faulty, i.e. improperly soldered or connected.
* *Electrical insulation* *of electrical contacts*
  + Ensure that any exposed metallic wire ends are properly insulated for safety purposes
  + Coat sufficient amount of tape – typically used black electrical insulating tape – fully around metallic wire ends
  + Confirm that insulating tape does not disrupt the functioning of other electrical components for testing.

****

**Figure 1:** Modified KiCad EDA design schematic of in-lab test. Schematic is in terms of motor controller design.

****

**Figure 2:** Reference schematic of three-phase diode bridge rectifier. Terminals ~C and B- are used to retrieve a single Schottky barrier from the rectifier.

**IV. Testing Procedure**

1. Start assembling the circuit based on Figure 1. Make sure you have all components available
2. The fuse relay and power resistors are connected in series to the power supply and parallel to the dual battery packs (Lithium-Ion and LiFePO4)
3. The thermistor is in contact with each battery for two seconds to get temperature readings. Then the motor controller is wired directly to the thermocouple digital thermometer for our separate temperature monitoring system accordingly - for alternate temperature sensing, use a thermal IR camera
4. Before running the test, setup the multimeter connection point to probe points as close to the motor speed controller as possible (this is due to the voltage drop across the wires at higher amps)
5. Connect the motor controller with the computer by clicking “connect” on the right hand side of the VESC tool, then under the window on the left-hand side, select RT (real-time data) and on top, select the current tab to display the current-time plots for the battery and the motor controller
6. Then begin the test experiment by closing the relay in the circuit and manually adjusting *i\_motor* (motor current) at the bottom left corner of the VESC tool from 0 A to 7 A, until the point the battery current *i\_batt* reaches 5A in conjunction. Note that VESC software handles the temperature logging of the MOSFETs on the motor controller, and this can be found on the top under the temperature tab
   1. record the current readings of each battery pack circuit frequently and continue to monitor the temperature.

**V. Measurable Criteria**

The criteria for successful testing of the second prototype of the eGlider hybrid battery pack are as follows:

1. Diode preventing reverse saturation current from LiFePO4 batteries to Li-ion batteries
2. Temperature of batteries is at room temperature (~20°C-25°C), and does not overheat (≥ 60°C)
3. Current is being drawn from LiFePO4 batteries
   1. Current sensor displaying current drawn from LiFePO4 batteries
   2. Collect data on current across adjustable load (especially the current at the discharge current of 5A)
4. Log aforementioned currents graphically over time using VESC tool software, as well as voltage and other temperature data acquired
5. MOSFETs on A50S motor controller do not overheat during testing (< 70°C)

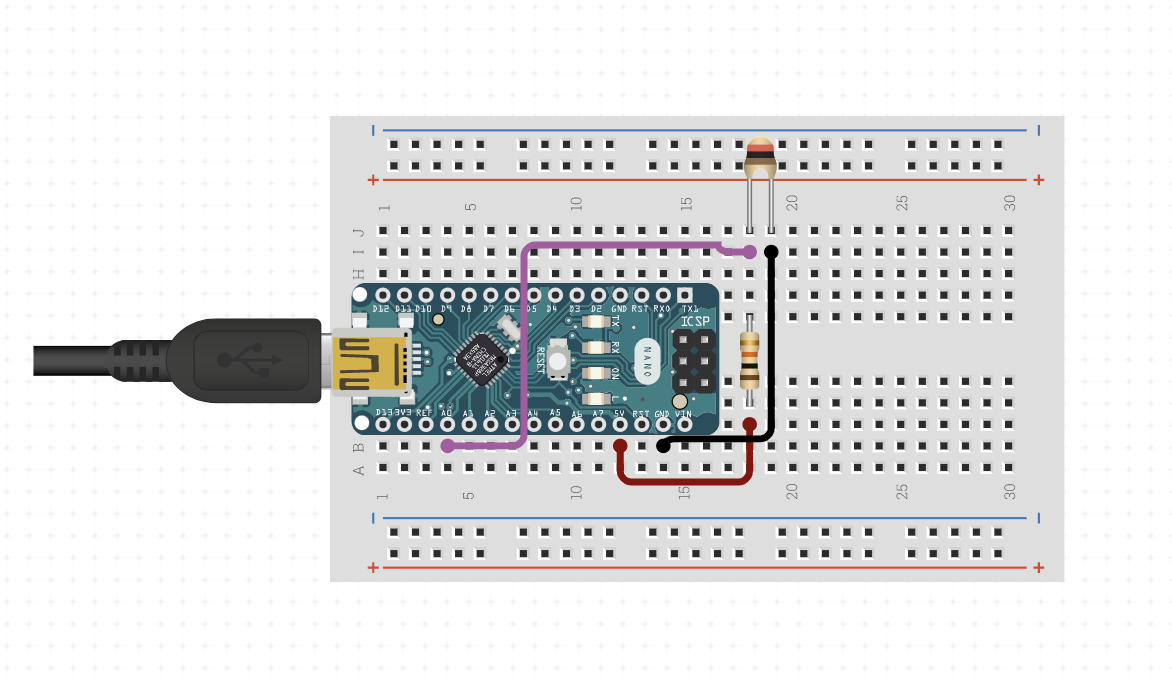
**VI. Score Sheet**

| **Motor Dischar-ge Current (A)** | **Li-ion Battery Voltage (V)** | **LiFePO4 Battery Voltage (V)** | **Battery Temperatu-res Li-ion (°C**) | **Battery Temperatu-res LiFePO4 (°C**) | **Motor Voltage using Multi-meter (V)** | **Battery Dischar-ge Current (A)** | **Motor Controller MOSFET Temp (< 70°C?)** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | **8.23** | **9.8** | **21.5** | **21.9** | **9.7** | **0.07** | **lower** |
| 2 | **8.23** | **9.71** | **21.5** | **21.9** | **9.6** | **0.36** | **lower** |
| 3 | **8.23** | **9.64** | **21.5** | **21.9** | **9.5** | **0.79** | **lower** |
| 4 | **8.23** | **9.53** | **21.5** | **21.9** | **9.4** | **1.39** | **lower** |
| 5 | **8.23** | **9.37** | **21.5** | **21.9** | **9.3** | **2.30** | **lower** |
| 6 | **8.23** | **9.15** | **21.5** | **21.9** | **9.0** | **3.42** | **lower** |
| 7 | **8.23** | **8.93** | **21.5** | **22** | **8.8** | **4.82** | **lower** |
| 8 | **8.23** | **8.63** | **21.5** | **22.2** | **8.5** | **6.54** | **lower** |

**VII. Hardware Pinout for Temperature Logging Microcontroller Circuit**

| **Arduino Nano Pin #** | **Usage/Description** |
| --- | --- |
| 27 | 5V power -> end (a) of thermistor |
| 29 | GND -> end (b) of thermistor |
| 19 | A0/D14 -> end (a) of thermistor |

**VIII. Arduino Nano Breadboard Digital Schematic (for one thermistor)**

****

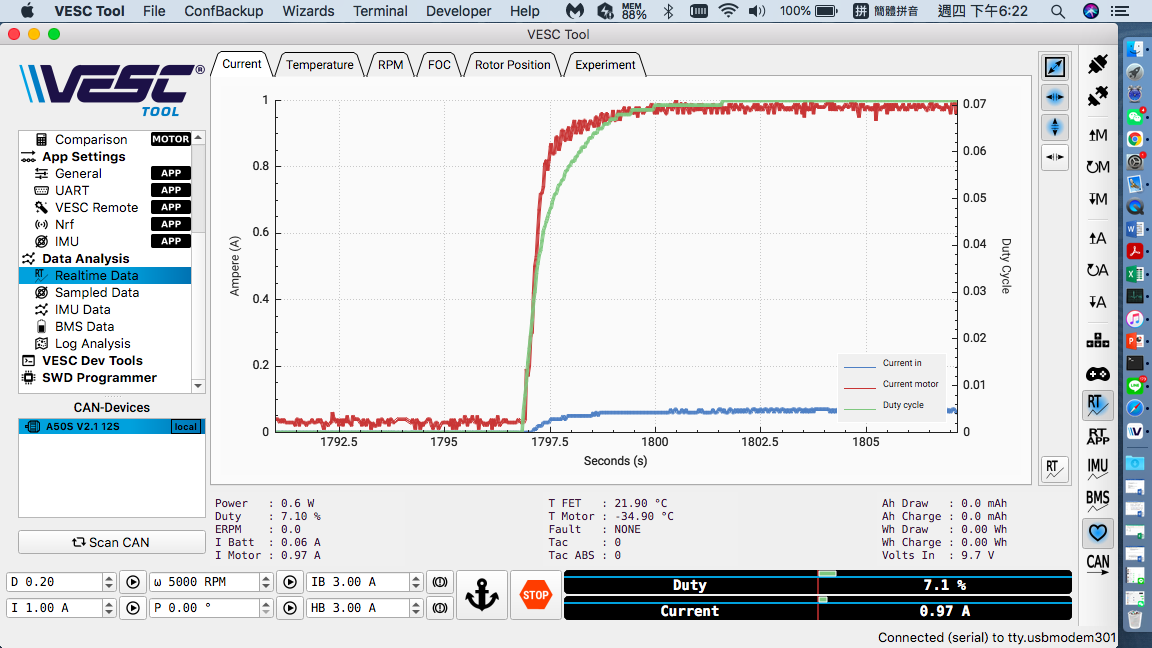
**Figure 3:** Digital schematic of Arduino Nano 33 BLE connections with 10K NTC thermistor and 10K resistor on solderless breadboard.

**IX. Measurements Taken**

The measurements taken during our tests are meant to show that our circuit is working correctly and all components are working properly.

1. **Li-ion Battery Voltage:** This is taken because we need to make sure that our diode is working properly.For our text the correct operation of the diode is expected to be reverse biased. We expected to see no voltage change in the Li-ion battery pack because it is providing no power to the motor controller. This is seen in our Score sheet above.
2. **LiFePO4 Battery Voltage:** This is taken because we need to know how many volts are going into the motor controller as well as the state of charge of our battery pack. Also ensures the LiFePO4 is the only battery pack providing power to the motor.
3. **Battery Temperatures Li-ion:** This is taken to ensure that we don’t overheat the Li-ion battery and they react to current as expected.
4. **Battery Temperatures LiFePO4:** This is taken because we need to ensure that we don’t overheat the LiFePO4 battery.
5. **Motor Voltage using Multimeter:** This is taken because we will need to see the voltage drop from our batteries to the motor controller.
6. **Battery Discharge Current:** This is taken because this can demonstrate how many amperes that we are pushing into the motor controller.
7. **Motor Discharge Current:** By adjusting the motor current in the VESC software we also adjust the current coming from the batteries.
8. **Motor Controller MOSFET Temperature:** This is taken because we need to ensure that we didn’t burn out the motor controller with such high ampere used.

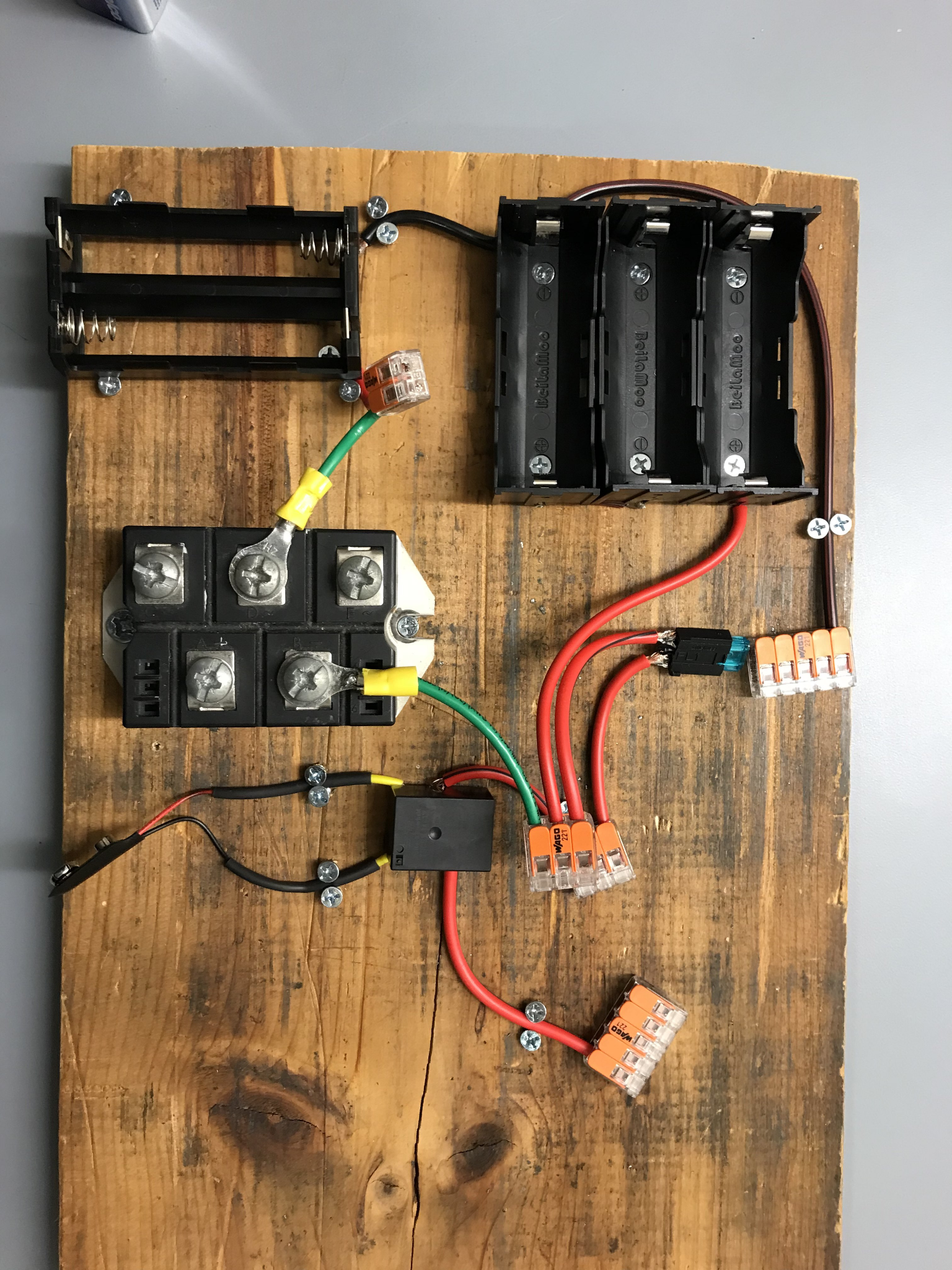
**X. Conclusion**

****

**Figure 4:** View of VESC Tool software during 1 A motor current test.

This test allows us to demonstrate that we are capable of controlling currents for future higher voltage and current tests more accurately than the previous tests. We were able to adjust the current for the motor which allows the current of the battery to reach 6.5 A. The reason why we can not go over 8.63 A for the motor current is because the load we are using is ~1 Ohm and the voltage is ~8.63 V. So by *V = IR* we cannot push more current without increasing our voltage. For the next test, there are two ways to increase the current. One way is to increase the numbers of the LiFePO4 batteries which allows us to gain higher voltage. Another way is to decrease the in-series resistance of the motor by making two 1 Ohm power resistors in halving the resistance and doubling our available current ceiling.

We learned some important lessons from this test. The first thing we learned is that we need to use slow blow fuses for these tests. The reason for this is that most motor controllers will have an input capacitor and if this capacitor is discharged while off – as they usually are when the connection to the batteries is made – the capacitor will cause an in-rush current large enough to blow our automotive grade fuses. The second lesson we learned is that for the sake of future safety and reliability, we need to create a more robust test bench such that the circuit components are stable and there isn’t a “rats’ nest” of wires that could cause a hazardous situation to arise such as a short circuit. As a result of these lessons learned we created a new testbench (**Figure 5**) which has shorter wires and is more electrically and mechanically robust and we are also researching slow blow inline fuses for future tests.



**Figure 5:** Updated test bench for test 4.